

## AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

### Listing of Claims:

1           1. (Currently amended) A method of automatically invalidating data cached in a  
2 cache system physically separate from a data server configured to originate data to be  
3 cached in the cache system, the method comprising:  
4           at a cache system physically separate from a data server configured to originate  
5 data to be cached in the cache system, caching a first data item received from the data  
6 server for service in response to requests to view said first data item;  
7           receiving at the cache system a change request from a client to alter a data item;  
8           comparing said change request to a set of rules for determining when to  
9 automatically invalidate said first data item; and  
10           automatically invalidating said first data item at the cache system in response to  
11 said change request.

1           2 (Canceled).

1           3. (Currently amended) The method of ~~claim 2~~claim 1, wherein said data item to  
2 be altered comprises said first data item.

1           4. (Original) The method of claim 1, wherein said automatically invalidating  
2 comprises identifying a relationship between said data item to be altered and said first  
3 data item.

1           5. (Original) The method of claim 1, wherein said cache system comprises  
2 multiple caches and said receiving occurs at a first of the multiple caches, the method  
3 further comprising:  
4           sending a notification of said automatic invalidation from the first cache of the  
5 multiple caches to a second cache of the multiple caches.

1           6. (Currently amended) A method of operating a cache system to facilitate  
2 automatic invalidation of cached data, the method being performed at the cache system  
3 and comprising:  
4           at a cache system physically separate from a data server configured to originate  
5 data to be cached in the cache system, caching a first set of data received from the a data  
6 server;  
7           receiving at the cache system a first request from a client to change a second set of  
8 data;  
9           retrieving from said first request an identifier of said second set of data;  
10          comparing said first request to a set of rules for determining when to  
11 automatically invalidate said first set of data; and  
12          automatically invalidating said first set of data in the cache system without  
13 awaiting an invalidation communication from the data server.

1           7. (Original) The method of claim 6, wherein said second set of data comprises  
2 said first set of data.

1           8. (Original) The method of claim 6, wherein said second set of data is a later  
2 version of said first set of data.

1           9. (Original) The method of claim 6, further comprising identifying a relationship  
2 between said second set of data and said first set of data.

1           10. (Original) The method of claim 9, wherein said identifying a relationship  
2 comprises:  
3           comparing a pattern of said first request to a first rule for determining when to  
4 automatically invalidate a set of data.

1           11. (Previously presented) The method of claim 10, wherein said first rule  
2 comprises a first pattern for identifying a request in response to which a set of data may  
3 be automatically invalidated, and further comprises a second pattern, different from said  
4 first pattern, to identify said set of data to be automatically invalidated.

1           12. (Original) The method of claim 6, wherein the cache system comprises  
2 multiple caches and:  
3           wherein said caching comprises caching said first set of data at a first cache of the  
4 multiple caches; and  
5           wherein said automatically invalidating is performed at a second cache of the  
6 multiple caches;  
7           the method further comprising notifying the first cache, by the second cache, of  
8 said automatic invalidation.

1           13. (Currently amended) A method of automatically invalidating cached data,  
2 comprising:  
3           caching a first set of data at a caching system for serving in response to a view  
4 request, wherein said view request comprises a request to view said first set of data, and  
5 wherein said first set of data is received from a data server;  
6           receiving at the caching system a change request from a client, wherein said  
7 change request comprises a request to change said first set of data;  
8           identifying said first set of data from said change request;  
9           comparing said change request to a set of rules for determining when to  
10 automatically invalidate said first set of data; and

11            automatically invalidating said cached first set of data without waiting for the data  
12            server to implement said change request, wherein invalidating said cached first set of data  
13            is in response to the view request.

1            14. (Previously presented) The method of claim 13, wherein said data server is  
2            notified of said change request only after said automatic invalidation of said cached first  
3            set of data.

1            15. (Previously presented) The method of claim 13, wherein said first set of data  
2            comprises a first user's bid on an item being auctioned in an electronic auction, and  
3            wherein said change request comprises a second user's bid on said item.

1            16. (Original) The method of claim 13, wherein said first set of data comprises  
2            information concerning one or more products at a publicly accessible network location,  
3            and wherein said change request comprises a change to said list of products.

1            17. (Original) The method of claim 13, wherein said receiving a change request  
2            comprises receiving a sequence of communications matching a predetermined pattern.

1            18. (Currently amended) A computer readable storage medium storing  
2            instructions that, when executed by a computing device, cause the computing device to  
3            perform a method of operating a cache system to facilitate automatic invalidation of  
4            cached data, the method being performed at the cache system and comprising:  
5                   at a cache system physically separate from a data server configured to originate  
6            data to be cached in the cache system, caching a first set of data received from the a data  
7            server;  
8                   receiving at the cache system a first request from a client to change a second set of  
9            data;  
10                   retrieving from said first request an identifier of said second set of data;

11        comparing said change request to a set of rules for determining when to  
12        automatically invalidate said first set of data; and  
13        automatically invalidating said first set of data in the cache system without  
14        awaiting an invalidation communication from the data server.

1        19. (Currently amended) A computer readable storage medium storing  
2        instructions that, when executed by a computing device, cause the computing device to  
3        perform a method of automatically invalidating cached data, the method comprising:  
4               caching a first set of data at a caching system for serving in response to a view  
5        request, wherein said view request comprises a request to view said first set of data, and  
6        wherein said first set of data is received from a data server;  
7               receiving at the caching system a change request from a client, wherein said  
8        change request comprises a request to change said first set of data;  
9               identifying said first set of data from said change request;  
10               comparing said change request to a set of rules for determining when to  
11        automatically invalidate said first set of data; and  
12               automatically invalidating said cached first set of data without waiting for the data  
13        server to implement said change request , wherein invalidating said cached first set of  
14        data is in response to the view request.

1        20. (Currently amended) A cache system configured to automatically invalidate  
2        cached data, comprising:  
3               a first cache configured to cache data received from a data server, wherein said  
4        first cache is separate from said data server, and wherein the data server is coupled to the  
5        cache system via a network link;  
6               a data service module configured to serve a first set of cached data in response to  
7        a first data view request from a client;  
8               a set of rules for determining when said first set of cached data is to be  
9        automatically invalidated in response to a data change request;

10 | a comparing module for comparing said first data change request to the set of  
11 | rules for determining when to automatically invalidate said first set of cached data; and  
12 | an invalidation module configured to automatically invalidate said first set of  
13 | cached data when a first data change request is received from a client;  
14 | wherein said automatic invalidation is performed at the cache system before the  
15 | data server is notified of said first data change request.

1 21. (Original) The cache system of claim 20, wherein said first cache comprises  
2 said invalidation module.

1 22 (Canceled).

1 | 23. (Currently amended) The cache system of ~~claim 22~~claim 20, further  
2 | comprising a user interface configured to facilitate the creation of one of said rules.

1 24. (Original) The cache system of claim 20, wherein said first data change  
2 request comprises a predetermined sequence of communications.

1 25. (Original) The cache system of claim 20, further comprising:  
2 a second cache;  
3 wherein said second cache is notified by said first cache of said automatic  
4 invalidation.

1 26-27 (Canceled).